IN THE CLAIMS

Please amend the claims as follows:

1. (Currently Amended) A method for managing multiple queues, comprising:

monitoring status for jobs in a first queue on a first system;

monitoring status for jobs in a second queue on a second system different than the first system, the first system and second system being located on different independently operating computing devices connected together through an Internet network or Local Area Network (LAN); and

managing the jobs in the first queue and the second queue from a same queue manager.

2. (Currently Amended) A method according to claim 1 wherein the first system is a network server coupled to the network and the second system is a peripheral device coupled to the network and including:

monitoring both the first queue on the network server and the second queue on the peripheral device remotely through the network so that jobs for the first and second queue appear together as a single displayed list of queue jobs;

receiving user job requests that effect the status of jobs in both the first queue in the network server and the second queue in the peripheral device; and

automatically controlling the jobs in both the first queue and second queue through commands sent over the network so that user job requests are conducted in conjunction with both the first queue and second queue at the same time.

3. (Original) A method according to claim 1 including displaying the jobs from the first queue and the second queue on a same user interface.

4. (Original) A method according to claim 3 including displaying on the user interface which of the first queue or the second queue is storing the different jobs.

5. (Currently Amended) A method according to claim 1 including:

receiving a select request to cancel one of the jobs;

identifying the first or second queue currently storing the selected job remotely over the network;

sending a cancel request over the network to the identified queue;

removing an identifier for the selected job from the queue manager when a confirmation is received over the network;

sending a cancel request over the network to the other one of the first or second queue when a cancel failure is received over the network from the identified queue; and

removing an identifier for the job from the queue manager when a confirmation is received over the network from the other one of the first and second queue.

6. (Currently Amended) A method according to claim 1 including:

receiving a select request to cancel one of the jobs;

not knowing which of the first and second queue currently contains the job associated with the select request and accordingly sending a cancel request to both the first and second queue; and

removing an identifier for the selected job from the queue manager when a cancel confirmation is received from either the first system or the second system.

7. (Currently Amended) A method according to claim 1 including:

receiving a selection request to change priority for one of the jobs;

~~identifying~~ sending a same query over the network to both the first and second system to determine which of the first or second queue is storing the selected job;

sending a request to change priority of the selected job to the identified queue; and

changing the priority identified for the job when a priority confirmation is received from the identified queue.

8. (Currently Amended) A method according to claim 7 including:

receiving a request to move priority for the selected job on the first queue above priorities for other jobs stored on the second queue;

sending a request over the network holding all jobs on the second queue having a priority below the priority requested for the selected job; and

releasing the jobs on hold when a confirmation is ~~receive~~ received over the network from the second queue that the selected job has been promoted on the second queue.

9. (Original) A method according to claim 7 including:

receiving a request to move the selected job on the first queue to a priority above other jobs stored on the second queue;

creating a slot in the second queue for the selected job; and

moving the selected job to the slot in the second queue.

10. (Currently Amended) A method according to claim 1 including:

receiving a request to demote a selected job on one of the first or second queue;

placing a hold on the identified job;

identifying all jobs having higher priority than the selected job in both the first and second queue; and

removing the hold on the selected job after all the identified higher priority jobs in both the first and second queue have been output.

11. (Currently Amended) A computer for providing queue management, comprising

a processor adapted to remotely monitor status of a server queue in a network server by sending server queue query messages over an Internet network or Local Area network and remotely monitor status of a device queue in a peripheral device by sending separate peripheral queue query messages over the network, the network server and peripheral device being separate devices operating at different locations on the network; and

a user interface adapted to display and manipulate the status of jobs in the first queue and second queue at the same time remotely over the network.

12. (Original) A computer according to claim 11 wherein the processor receives a request from the user interface to cancel a job and sends a cancel request to the server queue or device queue storing the job.

13. (Currently Amended) A computer according to claim 12 wherein the processor removes the job from a single list of jobs displayed on the user interface when a confirmation is received from the server or device queue that the job is cancelled.

14. (Original) A computer according to claim 12 wherein the processor automatically sends a cancel request to the device queue when a cancel request to the server queue fails.

15. (Currently Amended) A computer according to claim 11 wherein the device queue in the peripheral device and the server queue in the network server operate independently and the processor receives a request from the user interface to change priority for a job and then ~~controls scheduling of other jobs in~~ sends separate requests to both the server queue and the device queue according to the priority change request in order to synchronize the request with jobs in the server queue and device queue.

16. (Currently Amended) A computer according to claim 11 wherein the user interface displays a single list of multiple jobs for both the server queue and device queue waiting to be output, the output status of the jobs, a priority for outputting the jobs, and the server queue or device queue where the individual jobs are currently residing and automatically synchronizing the displayed list of multiple jobs by sending requests to each of the server queue and device queue for the same user job requests.

17. (Original) A computer according to claim 11 wherein the jobs can be any one of a fax job, print job, scan job, or copy job.

18. (Currently Amended) A computer according to claim 11 wherein the peripheral device can be any one or combination of the following:

a copier~~,~~;

a scanner;

a printer; or

a facsimile machine.

19. (Currently Amended) A system for managing jobs in queues, comprising:

a network server having a first queue for storing jobs;

a peripheral device having a second queue for storing jobs and outputting the jobs from the first queue and the second queue, the network server and the peripheral device being independently operating devices and the first queue and the second queue being independently operated queues that are not part of a same queuing device, the peripheral device receiving jobs from the first queue over a Local Area Network (LAN) network or Internet network; and

a queue manager coupled to both the network server and the peripheral device through the network for displaying and managing the jobs both on the network server and the

device though a same user interface so that jobs for the first and second queue appear together as a single displayed list of queue jobs, the queue manager receiving user job requests that effect the status of jobs in both the first queue and second queue and automatically managing the jobs in both the first queue and second queue to correspond with the user job requests.

20. (Currently Amended) A system according to claim 19 wherein the queue manager sends a request to the network server queue to cancel ~~of~~ or pause a job and then automatically sends a cancel or pause request to the device queue if the network server queue has already moved the job to the device queue.